Worksheet 4 Introduction to SQL Answers

Task 1

Conditions in SQL are constructed from the following operators:

Symbol	Meaning	Example	Notes
=	Equal to	CDTitle = “Autumn”	Different implementations use single or double quotes
>	Greater than	DatePublished > #01/01/2015#	The date is enclosed in quote marks or, in MS Access, # symbols.
<	Less than	DatePublished > #01/01/2015#	
<>	Not equal to	RecordCompany <> “ABC”	
>=	Greater than or equal to	DatePublished >= #01/01/2015#	
<=	Less than or equal to	DatePublished <= #01/01/2015#	
IN	Equal to a value within a set of values	RecordCompany IN (“ABC”, “DEF”)	
LIKE	Similar to	CDTitle LIKE “S*”	Finds titles beginning with “S” (wildcard operator varies and can be %)
BETWEEN… AND	Within a range, including the two values which define the limits	DatePublished BETWEEN #01/01/2015# AND #31/12/2015#	
IS NULL	Field does not contain a value	RecordCompany is NULL	
AND	Both expressions must be true for the entire expression to be judged true	DatePublished > #01/01/2015# AND RecordCompany = “ABC”	
OR	If either or both of the expressions are true, the entire expression is judged true.	RecordCompany = “ABC” OR RecordCompany = “DEF”	Equivalent to RecordCompany IN (“ABC”, “DEF”)
NOT	Inverts truth	RecordCompany NOT IN (“ABC”, “DEF”)	

The questions in Task 1 all relate to **tblFilm**, shown below.

FilmID	Title	Studio	ReleaseDate	ProductionCost($m)	BoxOffice($m)	Seen	Classification
1	Avatar	Fox	01 July 2009	254	2787.97	☑	12
2	Spider-Man 3	Sony	16 April 2007	286	890.87	☐	12
3	The Dark Knight Rises	WB	12 July 2012	230	1084.43	☐	12
4	The Hobbit: The desolation of Smaug	WB	13 December 2013	225	960.37	☑	12
5	Harry Potter and the half-blood Prince	WB	15 July 2009	268	934.42	☐	U
6	Pirates of the Caribbean:Dead Man's Chest	BV	24 June 2006	256	1066.18	☐	U
7	Shrek 2	DW	19 May 2004	100	919.83	☑	U
8	Pirates of the Caribbean: At world's end	BV	19 May 2007	300	963.42	☐	12
9	Skyfall	WB	23 October 2012	205	1108.56	☐	12
10	Titanic	Fox	19 December 1997	260	2186.77	☑	12

Write SQL statements to:

(a) select the Film ID, Title and Classification of all films with classification U or 12, which have been marked as "Seen".

The results should be ordered in Ascending order of Title.

```
SELECT FilmID, Title, Classification
FROM tblFilm
WHERE (Classification = "12" OR Classification = "U") AND (SEEN = True)
ORDER BY Title
```

(Could write SEEN = Yes, or SEEN = Y as alternatives to SEEN = True)

Which Film IDs will be selected, in what order?

Film IDs 1, 7, 4, 10

(b) Select the Title and Studio of all films released in 2012 or 2013 which took more than £220m at the box office.

```
SELECT Title, Studio
FROM tblFilm
WHERE (ReleaseDate between #01/01/2012# AND #31/12/2013#) AND
                    (BoxOffice($m)) > 220
```

(c) Select all columns for films from Fox, Sony or WB and display in descending order of release date

```
SELECT *
FROM tblFilm
WHERE Studio IN ("Fox", "Sony", "WB")
ORDER BY ReleaseDate DESC
```

or alternatively, WHERE Studio = "Fox" OR Studio = "Sony" OR Studio = "WB"

Task 2

This is a practical task in which you make the queries in an MS Access database.

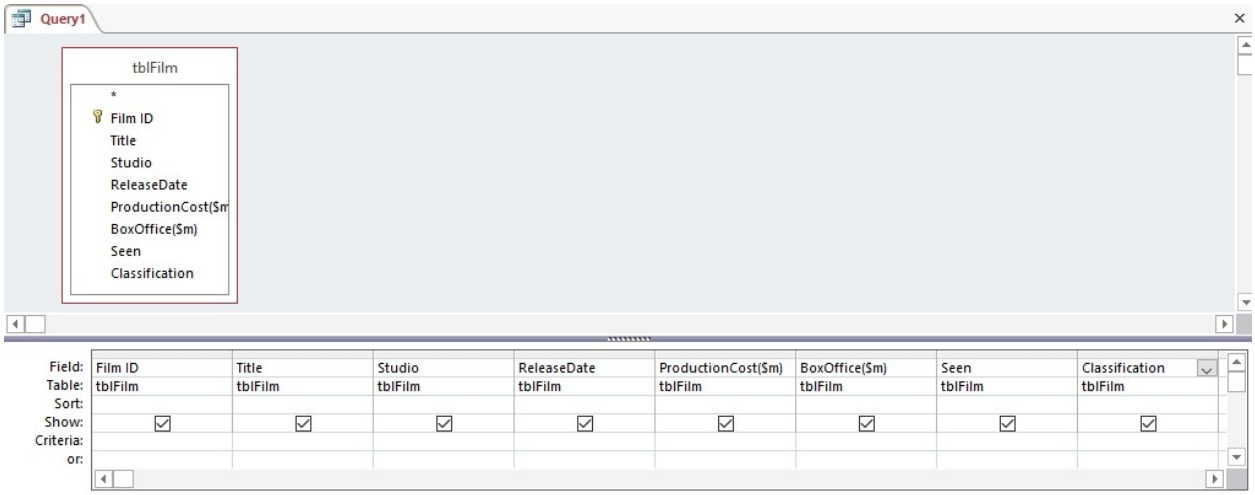
Viewing the database table

1. Load the database **Films4.accdb**. (This is in Access 2007 – Access 2013 compatible file format)

2. Double-click tblFilm to view the table tblFilm. It is the same table as the one displayed in Task 1.
3. Close the table by selecting the tab at the top of the table and selecting Close from the pop-up menu.

Creating a query

4. Select Create from the main menu, and then Query Design.
5. Add tblFilm in the Show Table window, and close the window.
6. In the Query window, double-click each field in turn to add them all to the query grid.



7. Righ-click the tab Query1 at the top of the query window and save the query as **Query1a**.
8. Look back at question 1a in Task 1. You are going to create this query. Deselect the fields that are not to be displayed, and write the conditions for Classification in the two rows “Criteria” and “or” , in the Classification column.
9. You will need to write the “Seen” criterion in both rows.
10. Click in the Sort row of the Title column and select Ascending.
11. Save the query again by pressing Ctrl-S.
12. Now run the query by clicking the Run icon, which looks like an exclamation mark (!)

    Your results will be displayed. Are they what you expected?

Viewing the SQL

Access generates its own SQL. Click the View icon (on the left of the Run icon) and select SQL view.

The SQL displayed is a rather long-winded version of the SQL that you wrote yourself, but it does the same job.

Task 3

The database **RevisionSubs.accdb** has three tables:

tblCustomer

custID	title	firstname	surname	email
C111	Mr	Fred	Carr	fcarr53@gmail.com
C245	Miss	Mabel	Jenkins	mabel777@bt.com
C364	Miss	Jasmine	Kumar	jkumar@icloud.com
C444	Mr	Basil	Brown	basil@brown.com
C501	Miss	Joanna	Kemp	jrkemp@rhs.sch.uk
C502	Mr	Stephen	Ross	seross@rhs.sch.uk
C503	Mr	Alan	Crabbe	ascrabbe@rhs.sch.uk
C513	Mr	Will	Kelly	wkelly2@mays.org.uk
C516	Miss	Emily	Grey	egrey@mays.org.uk
C520	Miss	Priti	Miah	pmiah@mays.org.uk

tblSubscription

subID	startDate	endDate	custID	productID
S1211	25/02/2016	24/02/2017	C111	p36
S1212	01/02/2016	31/01/2017	C111	p47
S1213	03/02/2017	03/02/2017	C245	p36
S1400	21/03/2016	20/06/2017	C444	p47
S1401	21/03/2016	20/06/2017	C444	p36
S1402	22/03/2016	21/03/2017	C501	p47
S1403	22/03/2016	21/03/2017	C502	p47
S1404	22/03/2016	21/03/2017	C503	p47
S1405	22/03/2016	21/03/2017	C503	p24
S1406	23/03/2016	22/03/2017	C513	p47
S1407	23/03/2016	22/03/2017	C520	p47
S1408	23/03/2016	22/03/2017	C516	p36

tblProduct

productID	productName	subject	level	price
p24	Equations	Maths	2	£12.00
p36	Programming	Comp Science	4	£5.00
p47	Database	Comp Science	4	£5.00

(a) List the IDs and surnames of all the customers who will be displayed by the following query:

```
SELECT tblCustomer.custID, firstname, surname, ProductName,
tblProduct.productID
FROM tblCustomer, tblProduct, tblSubscription
WHERE  tblCustomer.custID = tblSubscription.custID
    AND tblProduct.productID = tblSubscription.productID
    AND (productID = “p36” OR productID = “p24”)
```

C111 Carr

C245 Jenkins

C444 Brown

C503 Crabbe

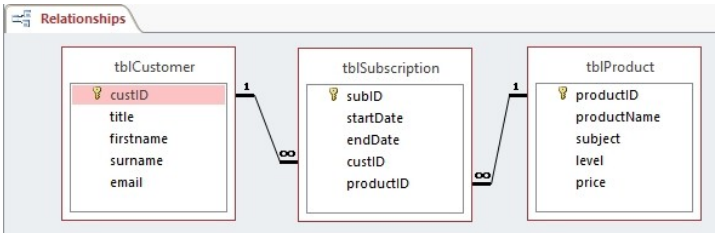
C516 Grey

(b) Write an SQL statement to display IDs and surnames all the customers at Mays School (identified by their email address) who have subscriptions for product p47.

```
SELECT tblCustomer.custID, tblCustomer.surname
FROM tblCustomer, tblSubscription, tblProduct
WHERE  tblCustomer.custID = tblSubscription.custID
    AND tblProduct.productID = tblSubscription.productID
    AND ProductID = “p47”
    AND email LIKE “*mays.org.uk”
```

Practical task

1. Open the database **RevisionSubs.accdb**.
2. Examine the tables, which are as shown in Task 2, and then close them.
3. Click Database Tools from the menu and select the Relationships icon.
4. Create the relationships as shown below. Remember to enforce referential integrity.



5. Now select Create from the menu, and click the Query Design icon. Add all three tables to the Query Design window.
6. Create the query in Question 2a using the Query By Example grid, as shown below.

Field:	custID	firstname	surname	productName	productID
Table:	tblCustomer	tblCustomer	tblCustomer	tblProduct	tblProduct
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:					"p36" Or "p24"
or:					

7. Run the query to check your answers to Question 2a.
8. Save the query as Query 2a.
9. View the query in SQL view. You will see the following:

```
SELECT tblCustomer.custID, tblCustomer.firstname, tblCustomer.surname,
    tblProduct.productName, tblProduct.productID
FROM tblProduct INNER JOIN (tblCustomer INNER JOIN tblSubscription ON
    tblCustomer.custID = tblSubscription.custID) ON tblProduct.productID =
    tblSubscription.productID
WHERE (((tblProduct.productID)="p36" Or (tblProduct.productID)="p24"))
```

10. Try creating a query by example corresponding to the query in Question 2b. (Use * not %)

    What results are displayed? C513, Kelly and C520, Miah

Field:	custID	surname	productID	email
Table:	tblCustomer	tblCustomer	tblProduct	tblCustomer
Sort:				
Show:	☑	☑	☐	☐
Criteria:			"p47"	Like "*mays.org.uk"